

日本国特許庁
JAPAN PATENT OFFICE

22.10.2004

別紙添付の書類に記載されている事項は下記の出願書類に記載されている事項と同一であることを証明する。

This is to certify that the annexed is a true copy of the following application as filed with this Office.

出願年月日 2003年11月18日
Date of Application:

出願番号 特願2003-388356
Application Number:
[ST. 10/C]: [JP2003-388356]

出願人 トヨタ自動車株式会社
Applicant(s):

REC'D 09 DEC 2004

WIPO

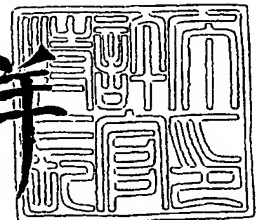
PCT

PRIORITY DOCUMENT
SUBMITTED OR TRANSMITTED IN
COMPLIANCE WITH
RULE 17.1(a) OR (b)

2004年11月26日

特許庁長官
Commissioner,
Japan Patent Office

小川 洋



BEST AVAILABLE COPY

【書類名】 特許願
【整理番号】 03-05862
【あて先】 特許庁長官殿
【国際特許分類】 G06F 9/46
【発明者】
 【住所又は居所】 東京都目黒区中目黒二丁目 1 番 2 3 号 株式会社トヨタ IT 開発
 センター内
 【氏名】 阿部 睦
【特許出願人】
 【識別番号】 000003207
 【氏名又は名称】 トヨタ自動車株式会社
【代理人】
 【識別番号】 100100549
 【弁理士】
 【氏名又は名称】 川口 嘉之
【選任した代理人】
 【識別番号】 100090516
 【弁理士】
 【氏名又は名称】 松倉 秀実
 【連絡先】 0 3 - 3 6 6 9 - 6 5 7 1
【選任した代理人】
 【識別番号】 100098268
 【弁理士】
 【氏名又は名称】 永田 豊
【選任した代理人】
 【識別番号】 100085006
 【弁理士】
 【氏名又は名称】 世良 和信
【選任した代理人】
 【識別番号】 100089244
 【弁理士】
 【氏名又は名称】 遠山 勉
【手数料の表示】
 【予納台帳番号】 192372
 【納付金額】 21,000円
【提出物件の目録】
 【物件名】 特許請求の範囲 1
 【物件名】 明細書 1
 【物件名】 図面 1
 【物件名】 要約書 1

【書類名】 特許請求の範囲**【請求項 1】**

タスク登録時に、特定リソースへアクセスする登録対象タスクについてその実行開始タイミングが遅延させられたとしてもそのデッドライン時間経過前に実行完了できるか否かを判定する判定部と、

そのデッドライン時間経過前に実行完了できると判定された登録対象タスクを実行対象タスクとして登録する登録部と、

タスク切替時に、前記実行対象タスクとして登録されたタスクのうち前記特定リソースへ先にアクセスしているタスクへのリソース割当が完了するまで、前記特定リソースへ後からアクセスしようとするタスクへのリソース割当を中止する制御部と、を備えるタスク管理システム。

【請求項 2】

タスク登録時に、特定リソースへアクセスする登録対象タスクについてその実行開始タイミングが遅延させられたとしてもそのデッドライン時間経過前に実行完了できるか否かを判定する判定部と、

そのデッドライン時間経過前に実行完了できると判定された登録対象タスクを実行対象タスクとして登録する登録部と、

前記実行対象タスクとして登録されたタスクのうち、デッドライン時間がより短いタスクを優先的に実行する実行部と、

タスク切替時に、前記実行対象タスクとして登録されたタスクのうち前記特定リソースへ先にアクセスしているタスクへのリソース割当が完了するまで、前記特定リソースへ後からアクセスしようとするタスクへのリソース割当を中止する制御部と、を備えるタスク管理システム。

【請求項 3】

タスク登録時に、特定リソースへアクセスする登録対象タスクについてその実行開始タイミングが遅延させられたとしてもそのデッドライン時間経過前に実行完了できるか否かを判定するステップ、

そのデッドライン時間経過前に実行完了できると判定された登録対象タスクを実行対象タスクとして登録するステップ、

タスク切替時に、前記実行対象タスクとして登録されたタスクのうち前記特定リソースへ先にアクセスしているタスクへのリソース割当が完了するまで、前記特定リソースへ後からアクセスしようとするタスクへのリソース割当を中止するステップ、を備えるタスク管理方法。

【請求項 4】

タスク登録時に、特定リソースへアクセスする登録対象タスクについてその実行開始タイミングが遅延させられたとしてもそのデッドライン時間経過前に実行完了できるか否かを判定するステップ、

そのデッドライン時間経過前に実行完了できると判定された登録対象タスクを実行対象タスクとして登録するステップ、

前記実行対象タスクとして登録されたタスクのうち、デッドライン時間がより短いタスクを優先的に実行するステップ、

タスク切替時に、前記実行対象タスクとして登録されたタスクのうち前記特定リソースへ先にアクセスしているタスクへのリソース割当が完了するまで、前記特定リソースへ後からアクセスしようとするタスクへのリソース割当を中止するステップ、を備えるタスク管理方法。

【書類名】明細書

【発明の名称】タスク管理システム

【技術分野】

【0001】

本発明は、実行可能なタスク間で各種入出力処理のような排他制御が必要なリソースへのアクセス競合から生ずる不具合を防止するための技術に関する。

【背景技術】

【0002】

従来、タスクの優先度等に応じてタスクを切り替えて実行する技術が知られている（例えば特許文献1参照）。

【特許文献1】特開2001-236236号公報

【発明の開示】

【発明が解決しようとする課題】

【0003】

しかしながら、実行可能なタスク間で各種入出力処理のような排他制御が必要なリソースへのアクセス競合が発生した場合、単純に従来技術のままタスクの切替を行うと、競合が発生しているので正しい処理が行えない。また、そのような状況を回避するため、アクセス競合が発生するリソースに単純に排他制御を行うと、想定外の遅延時間が発生し、設定されたデッドラインを守れない可能性がある。

【0004】

本発明の課題は、実行可能なタスク間で各種入出力処理のような排他制御が必要なリソースへのアクセス競合が発生した場合であっても、先にアクセスしたタスクへのリソース割当を保証し、且つ、それぞれのタスクへのリソース割当のデッドラインも保証するための技術を提供することにある。

【課題を解決するための手段】

【0005】

本発明は、上記課題を解決するためになされたものであり、タスク管理システムであって、タスク登録時に、特定リソースへアクセスする登録対象タスクについてその実行開始タイミングが遅延させられたとしてもそのデッドライン時間経過前に実行完了できるかを判定する判定部と、そのデッドライン時間経過前に実行完了できると判定された登録対象タスクを実行対象タスクとして登録する登録部と、タスク切替時に、前記実行対象タスクとして登録されたタスクのうち前記特定リソースへ先にアクセスしているタスクへのリソース割当が完了するまで、前記特定リソースへ後からアクセスしようとするタスクへのリソース割当を中止する制御部と、を備える構成とした。

【0006】

本発明によれば、タスク登録時に、特定リソース（例えば、実行可能なタスク間で各種入出力処理のような排他制御が必要なリソース）へアクセスする登録対象タスクについてその実行開始タイミングが遅延させられたとしてもそのデッドライン時間経過前に実行完了できるかを判定し、そのデッドライン時間経過前に実行完了できると判定された登録対象タスクを実行対象タスクとして登録する。そして、タスク切替時に、前記実行対象タスクとして登録されたタスクのうち前記特定リソースへ先にアクセスしているタスクへのリソース割当が完了するまで、前記特定リソースへ後からアクセスしようとするタスクへのリソース割当を中止する。従って、特定リソースへのアクセス競合が発生した場合であっても、先にアクセスしたタスクへのリソース割当を保証し、且つ、それぞれのタスクへのリソース割当のデッドラインも保証することが可能となる。

【0007】

また、本発明は次のように特定することもできる。

【0008】

タスク登録時に、特定リソースへアクセスする登録対象タスクについてその実行開始タイミングが遅延させられたとしてもそのデッドライン時間経過前に実行完了できるかを

を判定する判定部と、そのデッドライン時間経過前に実行完了できると判定された登録対象タスクを実行対象タスクとして登録する登録部と、前記実行対象タスクとして登録されたタスクのうち、デッドライン時間がより短いタスクを優先的に実行する実行部と、タスク切替時に、前記実行対象タスクとして登録されたタスクのうち前記特定リソースへ先にアクセスしているタスクへのリソース割当が完了するまで、前記特定リソースへ後からアクセスしようとするタスクへのリソース割当を中止する制御部と、を備えるタスク管理システム。

【0009】

また、本発明は方法の発明として次のように特定することもできる。

【0010】

タスク登録時に、特定リソースへアクセスする登録対象タスクについてその実行開始タイミングが遅延させられたとしてもそのデッドライン時間経過前に実行完了できると判定された登録対象タスクを実行対象タスクとして登録するステップ、タスク切替時に、前記実行対象タスクとして登録されたタスクのうち前記特定リソースへ先にアクセスしているタスクへのリソース割当が完了するまで、前記特定リソースへ後からアクセスしようとするタスクへのリソース割当を中止するステップ、を備えるタスク管理方法。

【0011】

タスク登録時に、特定リソースへアクセスする登録対象タスクについてその実行開始タイミングが遅延させられたとしてもそのデッドライン時間経過前に実行完了できると判定された登録対象タスクを実行対象タスクとして登録するステップ、前記実行対象タスクとして登録されたタスクのうち、デッドライン時間がより短いタスクを優先的に実行するステップ、タスク切替時に、前記実行対象タスクとして登録されたタスクのうち前記特定リソースへ先にアクセスしているタスクへのリソース割当が完了するまで、前記特定リソースへ後からアクセスしようとするタスクへのリソース割当を中止するステップ、を備えるタスク管理方法。

【0012】

また、本発明はプログラムの発明として次のように特定することができる。

【0013】

コンピュータに、タスク登録時に、特定リソースへアクセスする登録対象タスクについてその実行開始タイミングが遅延させられたとしてもそのデッドライン時間経過前に実行完了できると判定するステップ、そのデッドライン時間経過前に実行完了できると判定された登録対象タスクを実行対象タスクとして登録するステップ、タスク切替時に、前記実行対象タスクとして登録されたタスクのうち前記特定リソースへ先にアクセスしているタスクへのリソース割当が完了するまで、前記特定リソースへ後からアクセスしようとするタスクへのリソース割当を中止するステップ、を実行させるためのプログラム。

【0014】

コンピュータに、タスク登録時に、特定リソースへアクセスする登録対象タスクについてその実行開始タイミングが遅延させられたとしてもそのデッドライン時間経過前に実行完了できると判定するステップ、そのデッドライン時間経過前に実行完了できると判定された登録対象タスクを実行対象タスクとして登録するステップ、前記実行対象タスクとして登録されたタスクのうち、デッドライン時間がより短いタスクを優先的に実行するステップ、タスク切替時に、前記実行対象タスクとして登録されたタスクのうち前記特定リソースへ先にアクセスしているタスクへのリソース割当が完了するまで、前記特定リソースへ後からアクセスしようとするタスクへのリソース割当を中止するステップ、を実行させるためのプログラム。

【0015】

また、本発明は、上記プログラムを記録したコンピュータ読み取り可能な記録媒体としても特定できる。

【発明の効果】**【0016】**

本発明によれば、排他制御が必要なリソースのアクセスの有無の設定をタスク登録時に行い、排他制御による遅延を考慮し、タスク登録可能性判定を行うとともに、アクセス競合時にアクセス中のタスクへの割当が完了するまでアクセス競合が発生したタスクへの割当を中止することで排他制御が必要なリソースを使用する処理の正当性とデッドラインの保証が可能となる。

【発明を実施するための最良の形態】**【0017】**

以下、本発明の一実施形態であるタスク管理システムについて図面を参照しながら説明する。図1は、本実施形態のタスク管理システムの概略システム構成を説明するための図である。

【0018】

(システム環境)

本実施形態のタスク管理システムは、PDA(Personal Digital Assistants)やパーソナルコンピュータ等の一般的な情報処理装置により実現される。図1に示すように、この情報処理装置100は、CPU110、ハードディスク装置等の記憶装置120、特定リソース130、及びメモリ140等を備えている。また、各種情報や命令を入力するための入力装置(例えばキーセット)、その処理結果等を表示するための画像表示装置(例えば液晶ディスプレイ)、及び音声出力装置(例えばスピーカ)等も備えていることがある(いずれも図示せず)。

【0019】

(タスクとオペレーティングシステム)

タスク141は、プロセス、スレッドと一般的に呼ばれるプログラム実行単位の総称である。タスク141は、タスク登録時に生成されることもあるし、本出願人が先に出願したように予め生成(プール)されていることもある。タスク141は、例えば、その所要リソース量を制御可能な可変タスクであるQoSタスクである。

【0020】

オペレーティングシステム(OS)142は、各タスク141をDM(Deadline Monotonic)法によりスケジューリングする機能を有する(スケジューラ)、例えば、リアルタイムOSである。各スケジューリングタイミングにおいて、実行可能なタスク141(本発明の実行対象タスクに相当)の中から、最も短いデッドライン時間を持つタスク141がアクティブタスクとされる。各タスク141はタスク情報テーブル143により管理される。タスク情報テーブル143は、各タスク141に関する情報を管理するためのテーブルであり、タスク141に関する情報として、タスク識別子143a、起動タイミング143b、実行割当時間143c、デッドライン時間143d、及びリソース識別子143e等が登録される。

【0021】

タスク識別子143aは、各タスク141を識別するためのものである。起動タイミング143bは、各タスク141の実行開始時点から次の実行開始時点までの時間(周期)である。あるタスク141が実行されると、その周期の間、このタスク141が新たに実行されることはない。実行割当時間143cは、各タスク141に割り当てられるリソース量(ここでは、CPU110の使用時間)である。なお、一度リソースが割り当てられたタスク141が割当時間分だけ連続してリソースを割り当てられるとは限らない。何度かに分けて、時間を割り当てられることがある。また、あるタスク141よりも優先順位の高い別のタスク141によりプリエンプションされるとそのタスク141の処理が中断される。

【0022】

ある時間内に割当時間が経過すればシステムに影響を与えない。この時間がデッドライン時間143dである。本実施形態では、デッドライン時間143dが短いタスク141

の方が長いタスク 141 に比べて優先度が高く、デッドライン時間 143 d が短いタスク 141 (すなわち優先度が高いタスク) がアクティブタスクとされる。リソース識別子 143 e は、各タスク 141 が使用する排他制御が必要なリソース (本発明の特定リソースに相当) を識別するためのものである。このようなリソースとしては、プリンタやディスプレイ等がある。これらは、情報処理装置 100 に内蔵されていることもあるし、外付けされていることもある。なお、リソース識別子 143 e は、各タスク 141 に対して一つのことでもあるし、複数のこともある。

【0023】

上述のスケジュール機能や以下に説明するその他の諸機能等は、これら機能を提供する API (Application Program Interface) 等の所定プログラムが上述の情報処理装置 100 に読み込まれ、オペレーティングシステム 142 に組み込まれることにより実現される。なお、これらオペレーティングシステム 142 や所定プログラム等は、記憶装置 120 等にインストールされており、必要に応じて適宜メモリ 140 に読み出されて実行される (図 1 参照)。

【0024】

(タスク登録時動作)

次に、上記構成のタスク管理システムの動作について図面を参照しながら説明する。まずは、タスク登録時の処理について説明する。図 2 は、タスク登録時の処理について説明するための図である。

【0025】

以下の処理は、情報処理装置 100 にオペレーティングシステム 142 等が読み込まれて実行されることにより実現される。オペレーティングシステム 142 や所定アプリケーション等から新規タスク登録要求があった場合 (S100)、その登録要求のあったタスク 141 (本発明の登録対象タスクに相当。以下、登録対象タスクともいう。) が排他なリソース 130 (本発明の特定リソースに相当) を使用するか否かが判定される (S101)。例えば、その登録要求のあったタスク 141 に対応するリソース識別子 143 e と予め登録されている特定の識別子 (例えばプリンタやディスプレイ等の識別子。図示せず。) とを照合し、その結果一致した場合、その登録要求のあったタスク 141 は排他なリソース 130 を使用すると判定される (S101: Yes)。なお、その登録要求のあったタスク 141 は排他なリソース 130 を使用しないと判定された場合 (S101: No)、その登録要求のあったタスク 141 を新規タスクとして登録する (S104)。この場合、後述のタスク登録可能性の判定 (S102 および S103) は行わない。これは、その登録要求のあったタスク 141 を登録したとしても排他なリソース 130 へのアクセス競合が発生することがないためである。

【0026】

一方、その登録要求のあったタスク 141 は排他なリソース 130 を使用すると判定された場合 (S101: Yes)、タスク登録可能性が判定される (S102 および S103)。これは、その登録要求のあったタスク 141 のデッドライン時間 143 d を保証するためである。すなわち、そのタスク 141 を登録すると、排他なリソース 130 へのアクセス競合が発生し、デッドライン時間 143 d を守れない可能性があることから、タスク登録時に (事前に)、排他なリソース 130 へアクセスする登録対象タスク 141 についてその実行開始タイミングが遅延させられたとしてもそのデッドライン時間 143 d 経過前に実行完了できるか否かを判定しようとするものである (本発明の判定部に相当)。デッドライン時間 143 d 経過前に実行完了できないと判定された場合 (S103: No)、その登録要求のあったタスク 141 は登録されず、登録不許可通知がなされる (S105)。

【0027】

一方、デッドライン時間 143 d 経過前に実行完了できると判定された (S103: Yes) 登録対象タスク 141 は実行対象タスクとして登録される (本発明の登録部に相当)。例えば、その登録対象タスク 141 のタスク識別子 143 a やその登録対象タスク 1

41が使用するリソースのリソース識別子143e等がタスク情報テーブル143に登録される。

【0028】

(タスク登録可能性の判定の考え方)

上述のように、タスク登録可能性の判定は、デッドライン時間143dを保証するための処理であり、具体的には、タスク登録時に、排他なリソース130へアクセスする登録対象タスク141についてその実行開始タイミングが遅延させられたとしてもそのデッドライン時間143d経過前に実行完了できるか否かを判定するものである。

【0029】

この考え方について図3及び図4を参照しながら説明する。各図は、オペレーティングシステム142により実行される2つのタスクAとタスクBを例示する。タスクAとBは排他関係にあるリソース(例えば同一のリソース)130を使用する。また、タスクBのデッドライン時間よりもタスクAのデッドライン時間の方が短い。すなわちタスクBよりもタスクAの優先度が高い。

【0030】

図3は、タスクAが先に実行可能(または、タスクAとBが同時に実行可能)となった状態を示す。この場合、タスクAの割当時間143c終了までタスクBは待たされる。これは、タスクBよりもタスクAの優先度が高いからである。この場合、従来のスケジューリングで対応可能である。

【0031】

図4は、タスクBが先に実行可能となった状態を示す。この場合、タスクBの割当時間143c終了までタスクAは待たされる。つまり、排他的なリソース130にアクセスするタスクA及びBについては、タスク切替時にアクセス競合が発生したとしても、先にアクセスしたタスクBへのリソース割当が完了するまでは、後からアクセスしようとしたタスクAへの割当は遅延される。しかしながら、単純に遅延するとすれば、それぞれのタスクへのリソース割当のデッドライン時間を保証できない可能性がある。そこで、先にアクセスしたタスクBへのリソース割当(例えば実行割当時間143c)を保証しつつ、後からアクセスしようとしたタスクAへの割当(例えば実行割当時間143c)を遅延させたとしても、それぞれのタスクへのリソース割当のデッドライン時間143dを保証する必要がある。このため、タスク登録可能性(または、制限チェック、予約判定ともいう)を判定する(S102およびS103)。すなわち、双方のタスクへのリソース割当がそれぞれのタスクのデッドライン時間143dを満たすかどうかを判定する。具体的には、タスク登録時に、排他なリソース130へアクセスする登録対象タスクについてその実行開始タイミングが遅延させられたとしてもそのデッドライン時間143d経過前に実行完了できるか否かが判定される。その判定の結果、そのデッドライン時間143d経過前に実行完了(デッドライン保証)できると判定された登録対象タスクは実行対象タスク(新規タスク)として登録される。

【0032】

このように、タスク登録前に(事前に)タスク登録可能性の判定を行うこととしたので、仮にアクセス競合が発生したとしても、それぞれのタスクへのリソース割当のデッドライン時間143dも保証されることとなる。

(タスク登録可能性の判定の詳細)

次に、図2に戻ってタスク登録可能性の判定の詳細について説明する。

【0033】

登録要求のあったタスク141は排他なリソース130を使用すると判定された場合(S101: Yes)、その排他なリソース130と同じリソース130を使用するタスクが抽出される(S102)。例えば、登録要求のあったタスク141のタスク識別子143aに対応するリソース識別子143eをキーとして、これと一致するリソース識別子143eが対応付けられているタスク141がタスク情報テーブル143から抽出される。次に、その抽出されたタスク141は全て条件式をクリアしているか否かが判定される(

S103)。これは上述の通り、タスク登録可能性を判定するための処理である。以下、この条件式の具体例について説明する。

【0034】

まず、第1例として、優先度の高い順にタスクA, B, C, D, Eがあり、タスクCとDが排他関係にある場合を考える。この場合、次の2つの条件を満たすか否かが判定される。これらの条件を満たすのであれば、登録要求のあったタスク（例えばタスクC）はタスク登録可能と判定される（S103: Yes）。

（条件1）従来のDM法判定で、A, B, C, D, Eがスケジューリング可能であること

。（条件2） $D(c) \geq E(c) + (E(d) - 1\text{Tick時間})$

【0035】

なお、 $T(i)$ は*i*番目のタスクの周期を、 $D(i)$ は*i*番目のタスクのデッドライン時間を、 $C(i)$ は*i*番目のタスクの割当時間を、 $E(i)$ は*i*番目のタスクの所要時間を、 N は所要時間を求めるタスクの番号を、 $\neg A$ はAの演算結果を切り上げることを、それぞれ示す。また、 $E(c) = C(c) + C(a) \times (\neg E(c)/T(a) \neg) + C(b) \times (\neg E(c)/T(b) \neg)$ 、 $E(d) = C(d) + C(a) \times (\neg E(d)/T(a) \neg) + C(b) \times (\neg E(d)/T(b) \neg)$ である。

【0036】

条件2の意味について図4を参照しながら簡単に説明する。図4において、タスクAが最も長く待たされるのは、タスクBへの割当を開始した後の次のスケジューリングのタイミング（スケジューラが動いたとき）にタスクAが実行可能となった場合である。スケジューリングのタイミングの間隔が1Tickという一定の間隔で行われるため、この値（1Tick）が条件2に表れている。

【0037】

より詳細には、DM (DeadlineMonotonic) 法による登録可能性判定では、よりデッドラインの短いタスクによるプリエンプションがあることを前提にしている。しかし、アクセス制限タスクは排他関係にあるタスクが実行中の場合には、より短いデッドラインを持つ場合でもプリエンプションはできない。すなわち、タスクが実行開始点に達しても、より長いデッドラインを持つ排他関係のあるタスクの実行終了-1Tick分の時間待たされることがある。また、このとき、排他関係にあるデッドラインの短いタスクの実行時間も考慮する必要がある。

【0038】

したがって、「1つの排他関係にある優先順位の低いタスクの実行時間-1」+「すべての排他関係にある優先順位の高いタスクの実行時間」+「自タスク実行時間」が自タスクのデッドラインに収まっていればスケジューリング可能であると考えられる。なお、それぞれのタスクは排他関係にないより短いデッドラインを持つタスクにはプリエンプションされる可能性があるので、タスクの実行時間はそのことを考慮しなければならない。ここでは、この時間のことをタスクの所要時間と定義する。なお、タスクの所要時間($E(n)$)は上述の公式で導かれる。

【0039】

次に、第2例として、優先度の高い順にタスクA, B, C, D, E, Fがあり、タスクAとC間とBとDとE間で排他関係がある場合を考える。この場合、次の5つの条件を満たすか否かが判定される。これらの条件を満たすのであれば、登録要求のあったタスク（例えばタスクE）はタスク登録可能と判定される（S103: Yes）。

（条件1）従来のDM法判定でA, B, C, D, E, Fがスケジューリング可能であること。

（条件2） $D(a) \geq E(a) + (E(c) - 1\text{Tick時間})$

（条件3） $D(b) \geq E(b) + (E(d) - 1\text{Tick時間})$

（条件4） $D(b) \geq E(b) + (E(e) - 1\text{Tick時間})$

（条件5） $D(d) \geq E(d) + E(b) + (E(e) - 1\text{Tick時間})$

なお、 $E(a) = C(a)$ 、 $E(b) = C(b) + C(a) \times (\neg E(b)/T(a) \neg)$ 、 $E(c) = C(c) + C(b) \times (\neg E(c)/T$

(b) \neg)、 $E(d) = C(d) + C(a) \times (\neg E(d) / T(a) \neg) + C(c) \times (\neg E(d) / T(c) \neg)$ 、 $E(e) = C(e) + C(a) \times (\neg E(e) / T(a) \neg) + C(c) \times (\neg E(e) / T(c) \neg)$ である。

【0040】

S102で抽出されたタスク141は全て条件式(上述の第1例又は第2例等に示す条件式)をクリアしているか否かが判定される。そして、各タスク141が全て条件式をクリアしていると判定された場合(S101: No)、その登録要求のあったタスク141を新規タスクとして登録する(S104)。

【0041】

(タスク切替時動作)

次に、タスク切替時の処理について説明する。図5は、タスク切替時の処理について説明するための図である。

【0042】

以下の処理も、情報処理装置100にオペレーティングシステム142等が読み込まれて実行されることにより実現される。タスク情報テーブル143に実行対象タスクとして登録されたタスク141のうち、デッドライン時間143dがより短いタスク141が優先的に実行される(本発明の実行部に相当)。そして、タスクスケジューリングによる切替タイミングが到来すると(本発明のタスク切替時に相当)、切替可能なタスク141が有るか否かが判定される(S200)。例えば、タスク情報テーブル143に、より優先度の高いタスク141が登録されているか否かが判定される。切替可能なタスク141が無いと判定された場合(S200: No)、以下の処理は行わない。一方、切替可能なタスク141が有ると判定された場合(S200: Yes)、現在実行されているタスク141が使用中の排他リソース130を使用しない別のタスク141が有るか否かが判定される(S201)。これは、例えば、タスク情報テーブル143のリソース識別子143eを基に判定される。そして、そのような排他関係にないタスク141があれば(S201: Yes) 実行タスクは切り替えられる(S202)。一方、そのようなタスク141がなければ(S201: No) 実行タスクは切り替えられない。すなわち、排他関係にあるタスク141があれば(S201: No) 実行タスクは切り替えられない。上記S200~202の処理はタスクスケジューリングによる切替タイミングが到来するごとに繰り返される。これにより、タスク切替時に、実行対象タスクとしてタスク情報テーブル143に登録されたタスク141のうち、排他なリソース(本発明の特定リソースに相当)130へ先にアクセスしているタスク141へのリソース割当が完了するまで、その排他的なリソースへ後からアクセスしようとするタスク141への割当は遅延(又は中止)される(S201: No)。これは、本発明の制御部に相当する。しかも、このように遅延させたとしても、上述のタスク登録可能性の判定(S102およびS103)により、それぞれのタスク141へのリソース割当のデッドライン時間も保証されることとなる。

【0043】

以上説明したように、本実施形態のタスク管理システムによれば、タスク登録時には、排他制御による遅延が予め考慮された形でタスク141の予約判定が行われ(図2等参照)、タスク切替時には、アクセス競合が発生しても先にアクセスしたタスク141へのリソース割当が完了するまでは、後からアクセス使用としたタスクへの割当は遅延される。

【0044】

従って、アクセス競合が発生しても、先にアクセスしたタスク141へのリソース割当(実行割当時間143c)が保証され、且つ、タスク141に予約判定時にアクセス競合による遅延を考慮されているので(図2参照)、それぞれのタスク141へのリソース割当のデッドライン時間143dも保証される。

【0045】

本発明は、その精神または主要な特徴から逸脱することなく、他の様々な形で実施することができる。このため、上記の実施形態はあらゆる点で単なる例示にすぎない。これらの記載によって本発明が限定的に解釈されるものではない。

【産業上の利用可能性】

【0046】

本発明によれば、排他制御が必要なリソースのアクセスの有無の設定をタスク登録時に
行い、排他制御による遅延を考慮し、タスク登録可能性判定を行うとともに、アクセス競
合時にアクセス中のタスクへの割当が完了するまでアクセス競合が発生したタスクへの割
当を中止することで排他制御が必要なリソースを使用する処理の正当性とデッドラインの
保証が可能となる。

【図面の簡単な説明】

【0047】

【図1】本発明の一実施形態であるタスク管理システムの概略システム構成を説明す
るための図である。

【図2】タスク管理システムのタスク登録時の動作を説明するためのフローチャート
である。

【図3】タスク登録可能性の判定の考え方について説明するための図である。

【図4】タスク登録可能性の判定の考え方について説明するための図である。

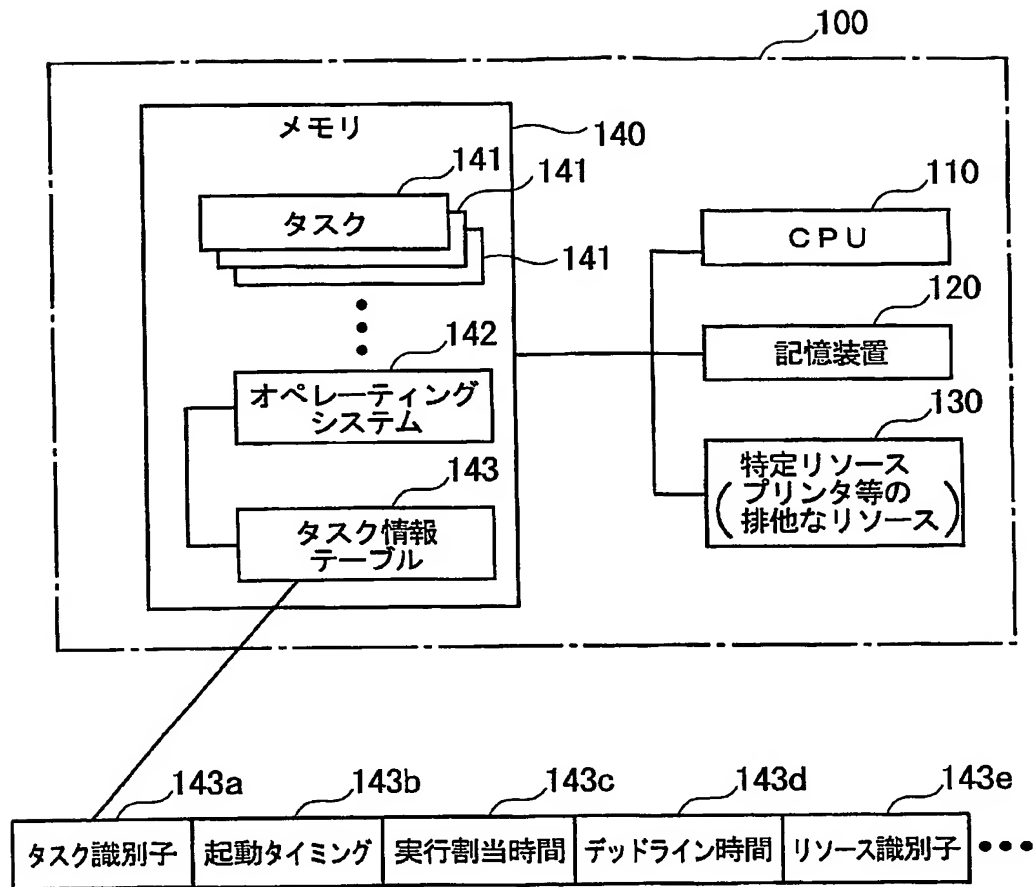
【図5】タスク管理システムのタスク切替時の動作を説明するためのフローチャート
である。

【符号の説明】

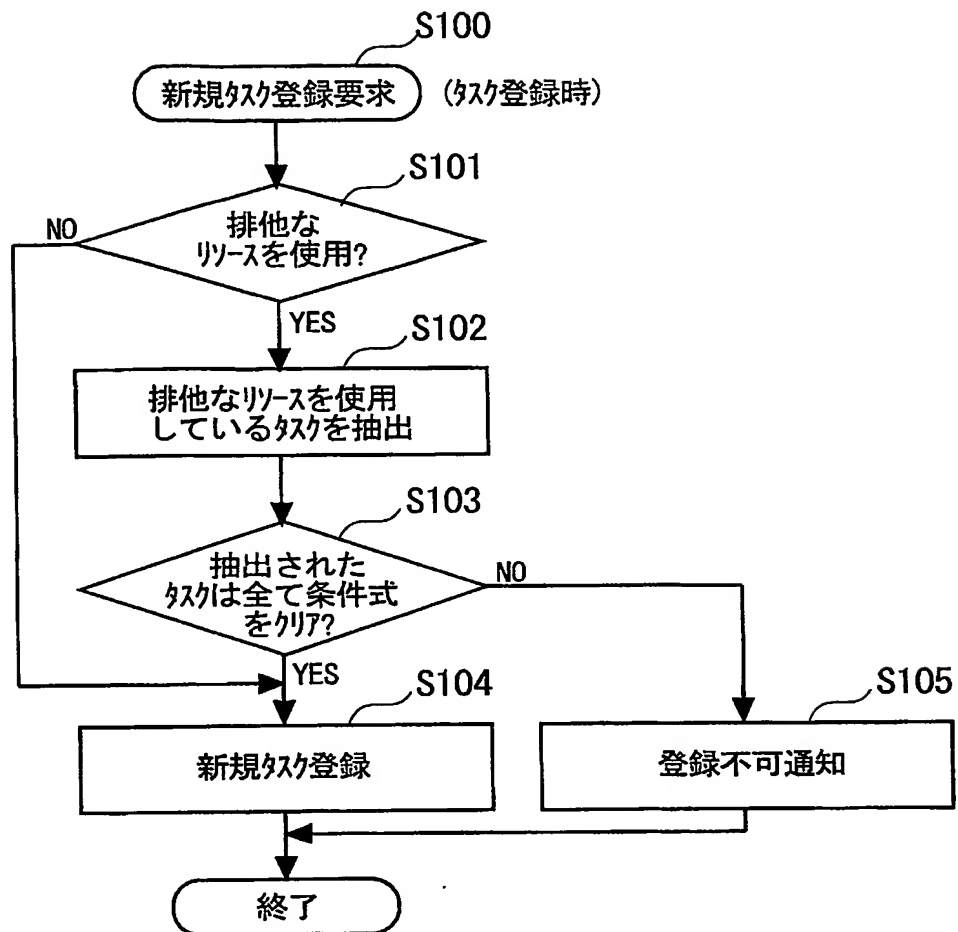
【0048】

100	情報処理装置
110	CPU
120	記憶装置
130	特定リソース
140	メモリ
141	タスク
142	オペレーティングシステム
143	タスク情報テーブル
143a	タスク識別子
143b	起動タイミング
143c	実行割当時間
143d	デッドライン時間
143e	リソース識別子

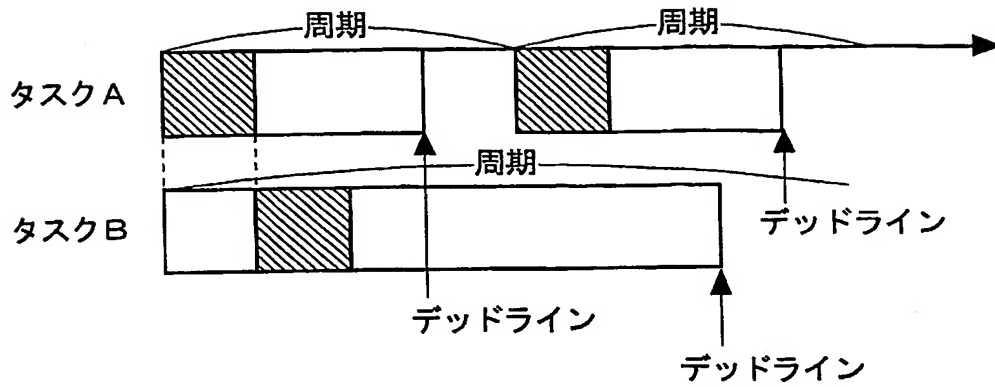
【書類名】 図面
【図1】



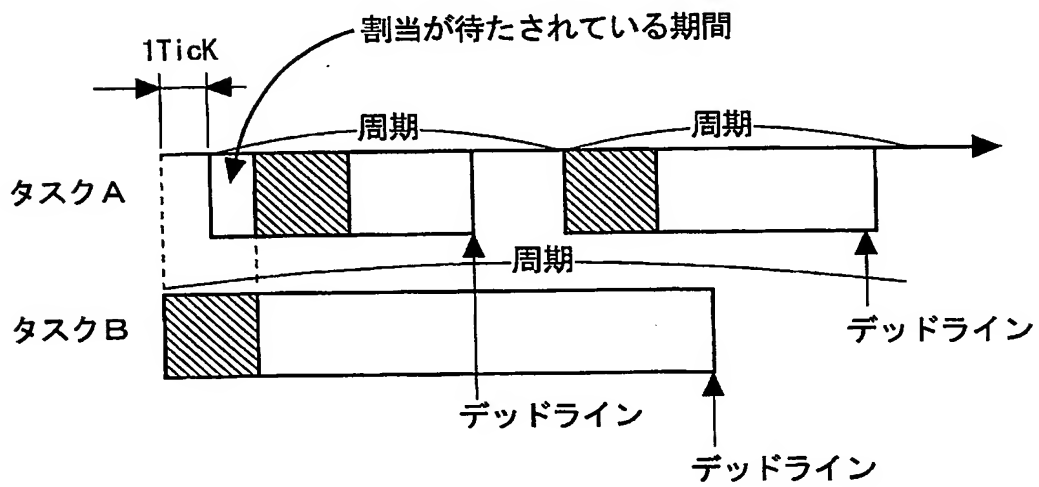
【図 2】



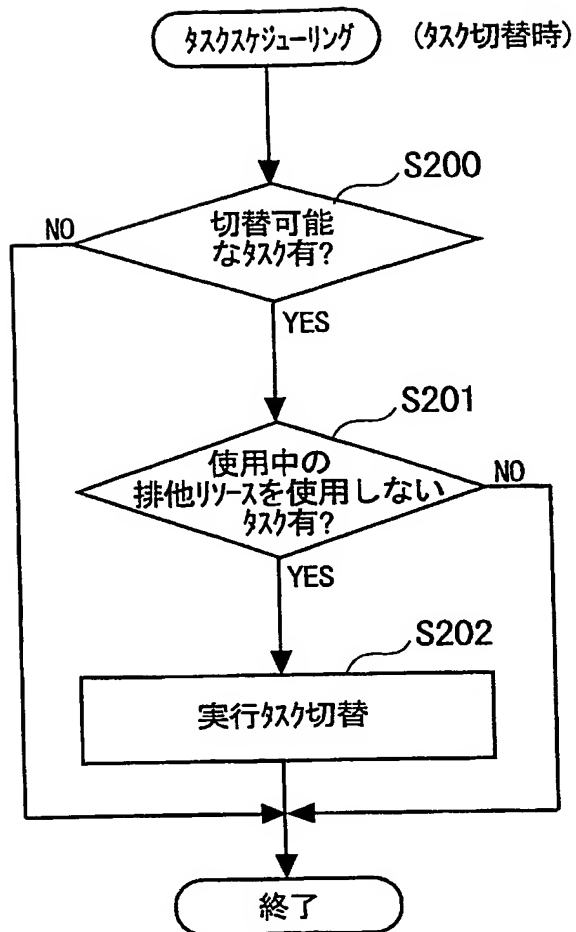
【図 3】



【図 4】



【図 5】



【書類名】 要約書**【要約】**

【課題】 実行可能なタスク間で各種入出力処理のような排他制御が必要なリソースへのアクセス競合が発生した場合であっても、先にアクセスしたタスクへのリソース割当を保証し、且つ、それぞれのタスクへのリソース割当のデッドラインも保証する。

【解決手段】 タスク登録時に、特定リソースへアクセスする登録対象タスクについてその実行開始タイミングが遅延させられたとしてもそのデッドライン時間経過前に実行完了できるか否かを判定する判定部と、そのデッドライン時間経過前に実行完了できると判定された登録対象タスクを実行対象タスクとして登録する登録部と、タスク切替時に、前記実行対象タスクとして登録されたタスクのうち前記特定リソースへ先にアクセスしているタスクへのリソース割当が完了するまで、前記特定リソースへ後からアクセスしようとするタスクへのリソース割当を中止する制御部と、を備える。

【選択図】 図2

認定・付加情報

特許出願の番号	特願 2003-388356
受付番号	50301905087
書類名	特許願
担当官	第七担当上席 0096
作成日	平成15年11月19日

<認定情報・付加情報>

【提出日】 平成15年11月18日

特願 2 0 0 3 - 3 8 8 3 5 6

出 願 人 履 歴 情 報

識別番号

[0 0 0 0 0 3 2 0 7]

1. 変更年月日

1 9 9 0 年 8 月 2 7 日

[変更理由]

新規登録

住 所

愛知県豊田市トヨタ町 1 番地

氏 名

トヨタ自動車株式会社

**This Page is Inserted by IFW Indexing and Scanning
Operations and is not part of the Official Record.**

BEST AVAILABLE IMAGES

Defective images within this document are accurate representations of the original documents submitted by the applicant.

Defects in the images include but are not limited to the items checked:

☒ **BLACK BORDERS**

☐ **IMAGE CUT OFF AT TOP, BOTTOM OR SIDES**

☐ **FADED TEXT OR DRAWING**

☐ **BLURRED OR ILLEGIBLE TEXT OR DRAWING**

☐ **SKEWED/SLANTED IMAGES**

☐ **COLOR OR BLACK AND WHITE PHOTOGRAPHS**

☐ **GRAY SCALE DOCUMENTS**

☒ **LINES OR MARKS ON ORIGINAL DOCUMENT**

☒ **REFERENCE(S) OR EXHIBIT(S) SUBMITTED ARE POOR QUALITY**

☐ **OTHER:** _____

IMAGES ARE BEST AVAILABLE COPY.

As rescanning these documents will not correct the image problems checked, please do not report these problems to the IFW Image Problem Mailbox.